Keyhole-aware laparoscopic augmented reality

Yamid Espinel[a,c,*], Navid Rabbani[a,c], Thien Bao Bui[b], Mathieu Ribeiro[c], Emmanuel Buc[c], Adrien Bartoli[a,c]

[a]*Université Clermont Auvergne, 28 place Henri Dunant, 63000 Clermont-Ferrand, France*
[b]*SURGAR Surgery, 22 allée Alan Turing, 63000 Clermont-Ferrand, France*
[c]*Centre Hospitalier Universitaire de Clermont-Ferrand, 58 rue Montalembert, 63000 Clermont-Ferrand, France*

ARTICLE INFO

ABSTRACT

Augmented Reality (AR) from preoperative data is a promising approach to improve intraoperative tumour localisation in Laparoscopic Liver Resection (LLR). Existing systems register the preoperative tumour model with the laparoscopic images and render it by direct camera projection, as if the organ were transparent. However, a simple geometric reasoning shows that this may induce serious surgeon misguidance. This is because the tools enter in a different keyhole than the laparoscope. As AR is particularly important for deep tumours, this problem potentially hinders the whole interest of AR guidance. A remedy to this issue is to project the tumour from its internal position to the liver surface towards the tool keyhole, and only then to the camera. This raises the problem of estimating the tool keyhole position in laparoscope coordinates. We propose a keyhole-aware pipeline which resolves the problem by using the observed tool to probe the keyhole position and by showing a keyhole-aware visualisation of the tumour. We assess the benefits of our pipeline quantitatively on a geometric in silico model and on a liver phantom model, as well as qualitatively on three patient data.

1. Introduction

Laparoscopic Liver Resection (LLR) has strong benefits compared to open hepatectomy Cheung et al. (2013). It is however not yet widely applicable Buell et al. (2009), covering about 20% of the procedures only. A major cause is the lack of reliable means of locating and visualising the intraparenchymal tumours and their oncologic resection boundaries. Laparoscopic Ultrasound (LUS) is the current gold standard but has limited depth range and is hindered by artefacts. Further, once the resection mark is done and the LUS probe removed, it may be difficult for the surgeon to mentally map the resection path from the US image to the laparoscopic image. Augmented Reality (AR) is a recent promising way to overcome these problems Plantefeve et al. (2015); Le Roy et al. (2019). The first step in AR is registration, which solves the non-rigid transformation that exists between the preoperative imagery, typically a CT volume, and the intraoperative laparoscopic images. Effective techniques have been recently proposed Plantefeve et al. (2015); Le Roy et al. (2019), based on anatomical landmarks, biomechanics and the Iterative Closest Point algorithm. The last step in AR is visualisation, which displays the registered data directly into the laparoscopic images.

The visualisation step is critical, for it must convey the information of internal structure location to enable gesture guidance. Surprisingly, the literature dealing with the visualisation step is relatively sparse. Some methods create visual effects to convey information such as oncologic tumour boundaries Espinel et al. (2020) and a depth scale Özgür et al. (2017) but most methods just blend the projected structures with the image Thompson

*Corresponding author: Tel.: +33-651-70-97-97;

e-mail: yamid@msn.com (Yamid Espinel), navid_rabbani@yahoo.com (Navid Rabbani), thienbaobuisn@gmail.com (Thien Bao Bui), mathieu.rgp@hotmail.fr (Mathieu Ribeiro), ebuc@chu-clermontferrand.fr (Emmanuel Buc), adrien.bartoli@uca.fr (Adrien Bartoli)
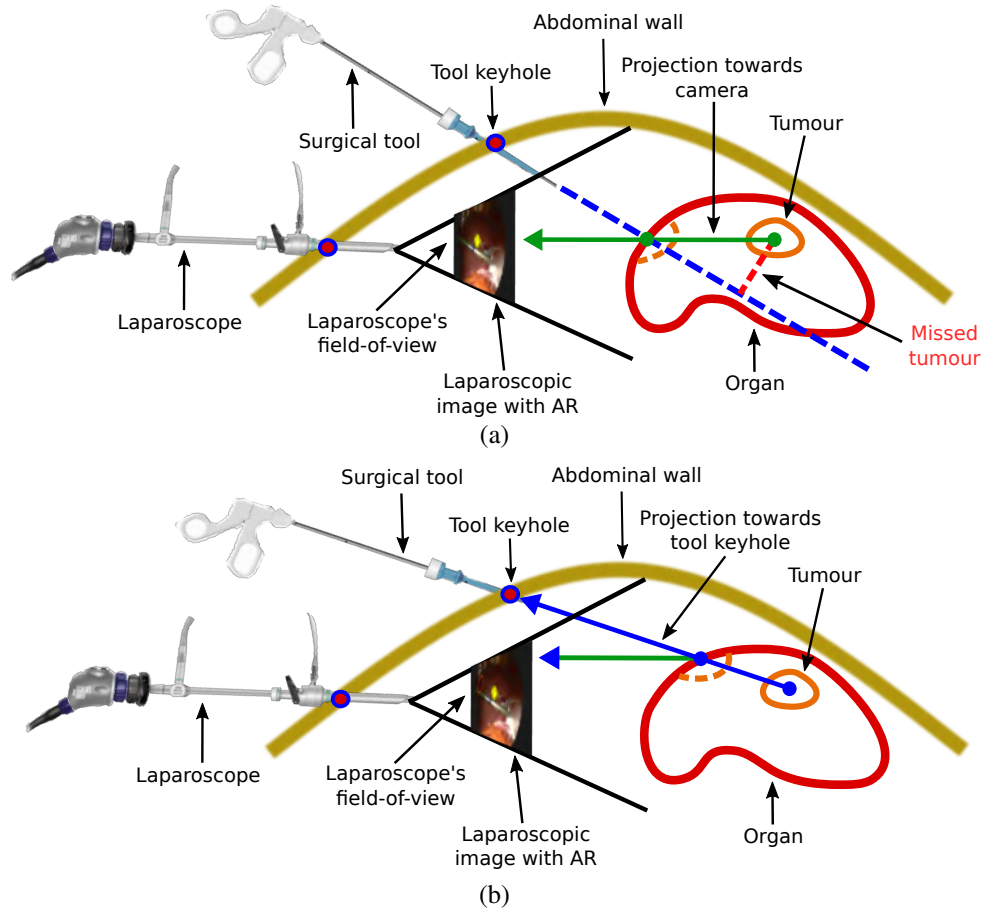
Fig. 1: Camera projection vs keyhole projection. (a) The existing direct camera projection methods do not allow the surgeon to aim at the tumour. (b) The proposed keyhole-aware projection method takes the tool keyhole into account, allowing the surgeon to directly aim following the AR indication.

et al. (2015); Bernhardt et al. (2015); Haouchine et al. (2014, 2016); Plantefeve et al. (2015); Clements et al. (2017); Chen et al. (2018); Robu et al. (2018); Koo et al. (2017); Adagolodjo et al. (2017). A complementary way to improve visualisation would be by adding a semi-transparent overlay of the liver surface on top of the tumour. This could help the surgeon perceive the virtual tumour inside the liver parenchyma, but does not bring additional information regarding its true location. The key common point of all these methods is to directly project the internal structures towards the laparoscope. This raises a critical problem when aiming at the tumour with the tools. This problem, shown in figure 1, comes from the fact that the tools are inserted through other keyholes different from the laparoscopic one, as identified in Collins et al. (2017). Quoting Collins et al. (2017): "In our user study we found this is a significant problem with smaller and/or deeper tumours, and can cause them to be missed." A simple and sensible solution named Tool Access Visualisation was proposed, which is to project the tumour to the anterior organ boundary towards the tool keyhole and only then to the laparoscope. The impact of this correction was not quantified, but is very important: a simple geometric reasoning shows that the typical aiming error could be of the order of 30 mm for deep tumours. This solution was tested in Collins et al. (2017) in an experimental setup involving an ex-vivo porcine

kidney and a pelvitrainer box. The organ was static and the tool keyhole position was physically measured prior to running the experiment. Tool Access Visualisation was never applied to a real clinical case, because it was not possible to estimate the tool keyhole position with respect to the organ. In summary, the limited field of view of LUS and the traditional camera-based projection used in AR make these methods unsuited to accurately localise deep tumours.

We propose a complete framework enabling the use of Tool Access Visualisation in the clinical setting in order to realise keyhole-aware AR. Our main objective is to improve the surgical gesture by means of an improved perception of the tumour's location. Specifically, we bring two key contributions. Our first contribution is a random sampling based algorithm that takes a set of unordered 3D tool poses and clusters them in keyholes while concurrently estimating the number of keyholes and their positions on the abdominal wall. Our second contribution is an algorithmic pipeline that enables the usage of keyhole-aware AR in laparoscopy. This pipeline estimates the 3D tool poses from the visible tools in the images, computes the keyhole locations using the proposed clustering algorithm, and projects the tumour's shape towards any of the estimated keyholes. This gives the surgeon a more accurate indication on how to reach the tumour using a tool inserted in a particular keyhole, com-

pared to a traditional camera-based projection which can misguide the surgical gesture, as shown in figure 1. This is a challenging problem, as the keyhole is never directly seen by the laparoscope, hence it must be localised by indirect means. Our method avoids the usage of third-party devices that would be necessary to estimate the keyhole locations, which are difficult to use in practice due to sanitary regulations. Our pipeline is fully automated and can be especially useful in three scenarios. First, when the tumour is not visible in the US image owing to its small size or important depth. Second, to have a view of the tumour's location directly on the laparoscopic image, rather than to mentally map it from the LUS image to the laparoscopic image. Third, for cirrhotic livers, which generally have many benign tumours and only a few malignant ones to be resected, which cannot be told apart using LUS, but can from the CT, and which can thus be properly indicated by AR. Although we focus on the tumour resection problem, our method can be used to display any internal structure such as blood vessels and bile ducts. We evaluate our method quantitatively on a geometric in silico model and a liver phantom model, where we assess both the tool clustering success and the keyhole estimation errors. From the estimated keyholes, we measure the pointing errors towards an intra-parenchymal tumour by using the traditional camera-based AR and the proposed keyhole-based AR. We also evaluate our method qualitatively on three patient data.

2. Methodology

2.1. General pipeline

Our system follows 6 steps, illustrated by figure 2. We assume the camera is static, which is done in practice using a Martin's arm holder. We calibrate the camera by filming a checkerboard and using OpenCV. In Step #1, the surgeon moves the tools around for a short time (under a minute) and the video stream is recorded. In Step #2, we detect geometric tool primitives (tool edgelines, midlines and shaft-tip) automatically in the recorded video images using ART-Net Hasan et al. (2021). In Step #3, we compute the tool 3D pose from the primitives using algebraic geometry Hasan et al. (2021). We save the tool axis L in camera coordinates for each image, which is represented by a point $q \in \mathbb{R}^3$ and an orientation vector $U \in \mathbb{R}^3$, $\|U\| = 1$. We thus obtain a set of tool 3D axes $\mathcal{L} = \{L_1, ..., L_n\}$. In Step #4, described in section 2.2, we find the keyhole locations $\mathcal{X} = \{x_1, ..., x_v\}$ by clustering the 3D axes. In Step #5, we register the preoperative 3D model to the images using Hepataug Koo et al. (2017). In Step #6, described in section 2.3, we visualise the tumours and projection cylinders with keyhole-aware projection based on the surgeon's keyhole choice.

2.2. Keyholes estimation and axes clustering

The process of estimating the keyhole locations works by clustering the tool 3D axes. Our clustering method iteratively finds the dominant cluster within the set of 3D axes $\mathcal{L}$. It proceeds by random sampling and has similarities with RANSAC Fischler and Bolles (1981). There are key differences however. First, RANSAC works for one cluster only. Second,

we give an improved sampling scheme, which preemptively discards erroneous axes from the axis set. Our approach is given in algorithm 1. Concretely, **KeyholesEstimationByAxesClustering** has a main loop to find the dominant cluster and keyhole position. It stores the keyhole position if the current dominant cluster is larger than a minimun cluster size s. The dominant cluster $\mathcal{S}$ is found by subroutine **FindDominantCluster**. This finds the cluster with maximum support from the axes and works by sampling keyhole positions and computing their support with subroutine **FindSupportingAxes**. An axis supports the keyhole hypothesis if their distance is lower than a constant f. Everytime a larger cluster is found, the number of required iterations u is updated, as in modern RANSAC implementations, from subroutine **MaxTrials**. The parameter u is computed according to equation (4.18) from Hartley and Zisserman (2004), as follows:

$$u = \frac{\log(1 - p)}{\log(1 - (1 - \epsilon)^s)}, \qquad (1)$$

where $p = 0.99$ is the probability of success and $\epsilon = 1 - \text{size}(\mathcal{S})/\text{size}(\mathcal{L})$ is the probability that any selected axis is an outlier. **FindDominantCluster** terminates by removing the clustered axes from the axis set. Finally, subroutine **SampleKeyholePosition** creates random keyhole hypotheses. It works by sampling pairs of axes and finding their mid-point, defined as their least-squares closest point. More specifically, our optimised sampling exploits the fact that if the mid-point is too far from the axes, then it is necessarily a false hypothesis, which does not require further consideration from the algorithm. For that, we first sample an initial axis using subroutine **SampleOneAxisRandomly**. We then randomly search for an axis which intersects up to a mid-point distance residual f, with subroutine **FindIntersectingAxisRandomly**. If such an intersecting axis is not found, it means that the initial axis can never be clustered, hence is an erroneous measurement not pertaining to any keyhole, and is thus removed from the axis set. The two global variables, namely the minimum number of axes s required to support a valid keyhole and the maximum clustering axis to keyhole distance f, are selected according to the inter-keyhole distance and the minimum number of reconstructed tool axes.

2.3. Keyhole-aware tumour and projection cylinder visualisation

Keyhole-aware visualisation is a sequence of two projections, as shown in figure 3: first, an orthographic projection of the internal structure to the liver surface; second, a perspective projection to the camera. Concretely, we implemented two visualisation modes: a simple mode where only the tumour is visualised, and an advanced mode where a projection cylinder is visualised. This projection cylinder is designed to help the surgeon to locate the tumour, to aim at it from the selected keyhole and to improve their depth perception. This cylinder is constructed in 3D from a line joining the keyhole to the tumour centroid and clipped to the parenchyma, hence corresponding to the first projection. We represent it graphically by a generalised cylinder whose axis is the projection line, following five steps. First, we generate a clipping plane C oriented towards
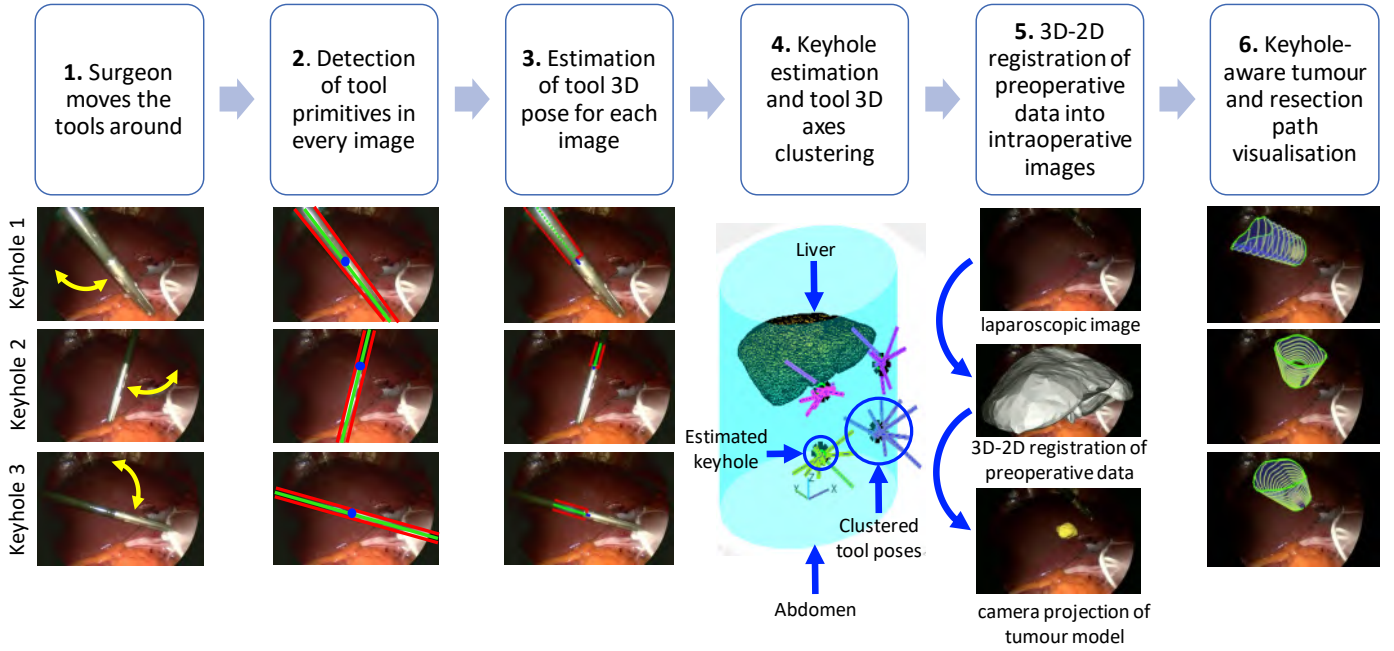
Fig. 2: Pipeline of the proposed keyhole-aware AR method on a patient case with three tool keyholes, shown as the three rows. In Step #2, the red lines correspond to the tool's edge lines, the green line correspond to the tool's middle line, and the blue point correspond to the tool's head tip. Once the keyhole positions are estimated (Step #1 to Step #4) and preoperative registration solved (Step #5) the surgeon can choose a keyhole from which to aim at the tumour (Step #6).

the keyhole and with origin at the tumour centroid. Second, we construct the tumour boundary contour as the intersection between the tumour, as obtained from the registered mesh, and the clipping plane. Third, we expand this contour by 1 cm within plane C to add the resection margin advised for Hepatocellular Carcinoma (HCC) interventions Zhong et al. (2017). The size of this margin can also be specified by the surgeon. Fourth, we project the margin curve orthographically towards the keyhole, creating the generalised cylinder. The intersection between the cylinder and the liver surface is used to draw the margin projection on the liver surface. Fifth, we sample the cylinder in planes parallel to plane C and 1 cm apart, providing depth visualisation. Finally, we use perspective projection to render the generated elements on the camera, forming the augmented laparoscopic image with the keyhole-aware tumour and the projection cylinder. Both the proposed simple and advanced keyhole-aware visualisation modes can be used with any projection direction chosen by the surgeon. However, projecting the tumour towards the tool keyhole allows the surgeon to directly aim at the tumour.

3. Experimental Setup

3.1. Geometric in silico model

We use the geometric 3D model shown in figure 4 to simulate a laparoscopic setup and measure the clustering, keyhole and pointing errors. Our model uses a cylinder to represent a human abdomen, with a radius $r = 16.5$ cm, which is a common size for an insufflated abdominal cavity Malbrain et al. (2014). We position a liver 3D model at a distance $g = 5$ cm from the abdominal wall and a virtual tumour represented by its centre

point t located 3 cm away from the liver surface. We use 4 keyholes distributed in a rectangular fashion with an inter-keyhole distance $k = 15$ cm and a laparoscope keyhole located beneath the tool keyholes.

Assessment of the clustering success and keyhole error. We use our geometric 3D model to randomly generate multiple tool axes passing through each of the keyholes. Because the real estimated tool axes are affected by detection errors, calibration errors and deformations of the abdominal wall, we add random noise to the generated axes. The tool axes are thus simulated as follows: for a given keyhole x, we sample a line $L = (x', W)$ where x' is obtained by perturbing x with random noise and W is a random unit vector. The perturbed keyhole is defined as $x' = x + dV$, where V is a unit vector perpendicular to W and d is the noise magnitude. This process is repeated for every generated tool axis. We create both valid and erroneous axes by changing the noise distribution. For valid axes, we use a normal distribution $d \sim \mathcal{N}(\mu_0, \sigma_0^2)$, with $\mu_0 = 0$ mm and $\sigma_0 = 5$ mm. The value of σ_0 corresponds to the median tool pose estimation error reported in Hasan et al. (2021). For erroneous axes, we use a uniform distribution $d \sim U(a, b)$, with $a = 1.965 \sigma_0$ and $b = 150$ mm. The value of a corresponds to the 97.5th percentile point of the normal distribution $\mathcal{N}$. We generate a total of $n = 1000$ tool axes, from which $e = 10\%$ correspond to erroneous axes. These axes are drawn at 40% for keyhole x_1, 30% for keyhole x_2, 20% for keyhole x_3 and 10% for keyhole x_4. We measure the clustering success and keyhole error against varying values of noise magnitude σ_0, number of observed tools n, and rate of erroneous tools e. The clustering success corresponds to the percentage of correctly detected tool axes for a given cluster $\mathcal{S}_i$ belonging to a groundtruth keyhole x_i. An axis is clustered to an estimated keyhole x' if it lies closer to it than
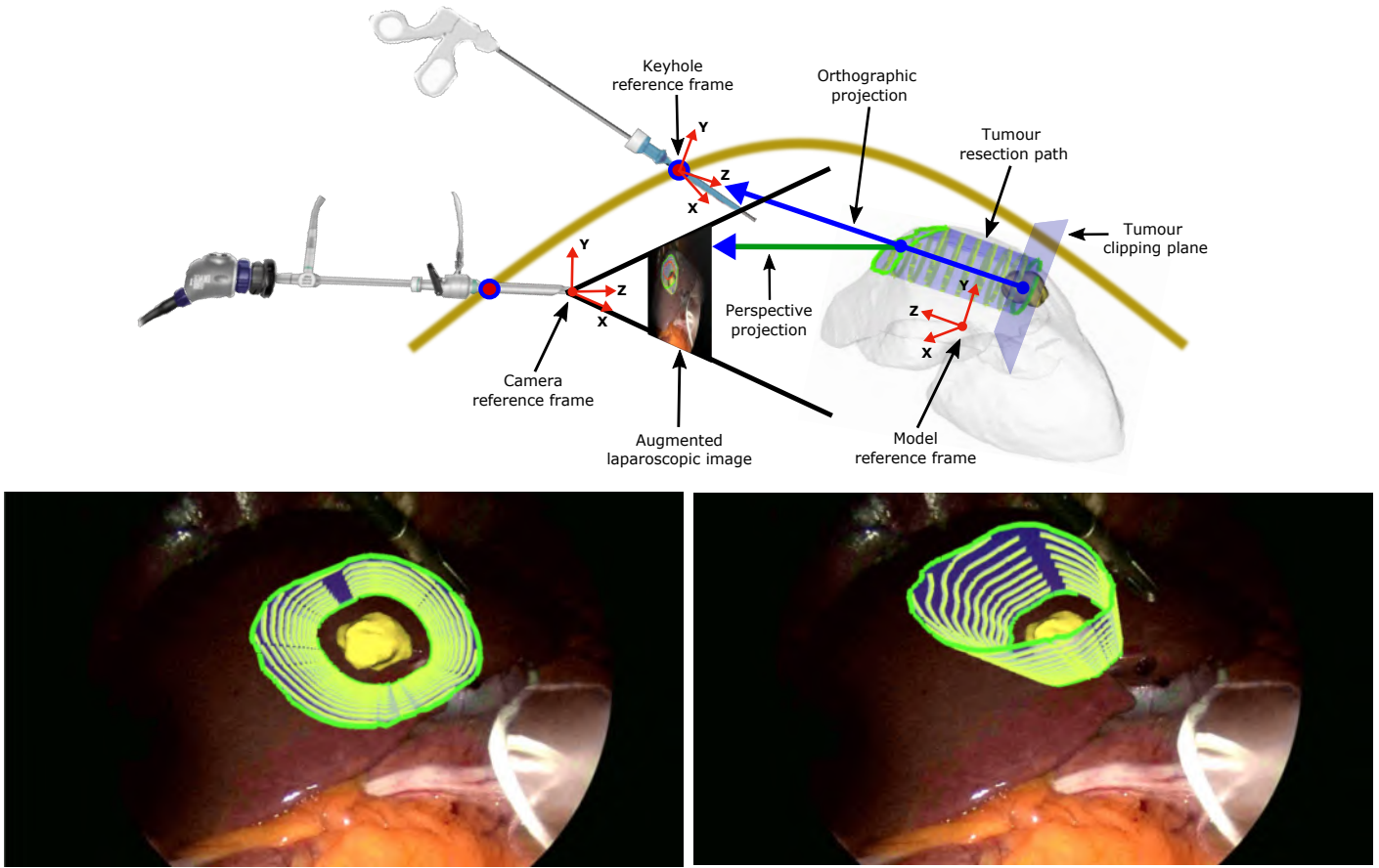
Fig. 3: Generation of keyhole-aware projection cylinder. (top) In keyhole-aware projection, an orthographic projection of the tumour boundary and the resection margin on the liver surface is first performed towards the keyhole. Then, a perspective projection of the generated projection cylinder is done towards the camera. (bottom) Projection cylinders for direct camera projection (left) and keyhole-aware projection (right). The projection cylinder joins the tumour occluding contour inside the parenchyma to the liver surface, following the direction of the camera (left) or of the keyhole (right). The green curve represents the 1 cm resection margin advised for HCC interventions. Each yellow curve represents a 1 cm distance through the parenchyma.

the maximum keyhole-to-axis distance f. A cluster is considered to be found if it is supported by at least the minimum number of tool axes s. Each cluster containing the largest number of tool axes corresponding to a groundtruth keyhole x_i among all the detected clusters is considered successful. We then proceed to measure the error between each cluster's estimated keyhole x' and the assigned groundtruth keyhole x_i. Therefore, the keyhole error is only estimated for successful clusters. The keyhole error is the euclidean distance between an estimated keyhole x'_i and the corresponding groundtruth keyhole x_i. We vary σ_0 from 0 to 20 mm in steps of 2 mm, n from 100 to 2000 in steps of 100, and e from 0% to 50% in steps of 5 mm. For every value of n and e, we use the same tool distribution of 40% for keyhole x_1, 30% for keyhole x_2, 20% for keyhole x_3 and 10% for keyhole x_4. At every step, we run the clustering algorithm 100 times with $s = 5$ and $f = 20$ mm. For each of the varying parameters, we keep the other two parameters fixed at the default values defined previously. We measure the error mean and standard deviation across all runs. The results of these experiments are shown in section 4.1.

Assessment of the pointing errors. We use our geometric 3D model to measure the influence of keyhole-aware visualisation, compared to direct camera projection, on tumour aiming with a

surgical tool, as illustrated in figure 5. To achieve this, we first generate a keyhole $x' = x + dW$ from one of the true keyholes x, where W is a random unit vector and $d \sim \mathcal{N}(\mu_0, \sigma_0^2)$ is the noise magnitude. From the results obtained in section 4.1 and the typical settings of laparoscopy, we set $\mu_0 = 1.9$ mm and $\sigma_0 = 0.45$ mm. We use keyhole x_1 in our experiments to measure the pointing errors. We define the pointing error as the distance between the tool axis and the tumour while pointing towards the direct-camera or the keyhole-aware projection of the tumour. The direct-camera pointing error, or type-A error, occurs when the surgeon follows the existing direct camera projection. We measure it by first projecting the tumour t to the liver surface S towards the camera centre c, giving point $w_c = (t, c) \cap S$ and then finding the distance between line (x, w_c) and tumour t. The keyhole-aware pointing error, or type-B error, occurs when the surgeon follows the proposed keyhole-aware visualisation. We measure it by first projecting the tumour t to the liver surface S towards the keyhole x', giving point $w_x = (t, x') \cap S$ and then finding the distance between line (x, w_x) and the tumour t. We measure the pointing errors against tumour depth between 0 and 100 mm in steps of 5 mm, inter-keyhole distance between 100 and 200 mm in steps of 5 mm, and keyhole precision between 1.84 and 4.2 mm in steps of 0.1 mm. We define the tumour

Function KeyholesEstimationByAxesClustering
  Input: $\mathcal{L} = \{L_1, ..., L_n\}, f, s$ // Set of 3D axes, maximum distance for valid supporting axis, minimum cluster size
  Output: $\mathcal{X} = \{x_1, ..., x_v\}$ // Set of 3D keyholes
  converged ← false, $\mathcal{X} ← \emptyset$
  while not converged **do**
    if size($\mathcal{L}$) $< s$ **then**
      converged ← true // Too few axes for a new cluster
    $[\mathcal{S}, x, \mathcal{L}]$ ← FindDominantCluster($\mathcal{L}, f, s$)
    if size($\mathcal{S}$) $< s$ **then**
      converged ← true // Dominant cluster too small for a new keyhole
    else
      $\mathcal{X} ← \mathcal{X} \cup \{x\}$ // Dominant cluster forms a new keyhole
    end
  end
Function FindDominantCluster
  Input: $\mathcal{L} = \{L_1, ..., L_m\}, f, s$ // Set of 3D axes, maximum distance for valid supporting axis, minimum cluster size
  Output: $\mathcal{S} = \{L'_1, ..., L'_h\}, x, \mathcal{L}$ // Set of 3D axes in the dominant cluster, 3D keyhole location, set of remaining 3D axes
  $u ← \infty$ // Maximum number of trials
  $k ← 0$ // Trial count
  $\mathcal{S} ← \emptyset$
  repeat
    $[\mathcal{L}, x]$ ← SampleKeyholePosition($\mathcal{L}, f, s$)
    $\mathcal{S}'$ ← FindSupportingAxes($\mathcal{L}$, x, s) // Find supporting axes for keyhole
    if size($\mathcal{S}'$) > size($\mathcal{S}$) // If new support is greater than current best **then**
      $\mathcal{S} ← \mathcal{S}'$ // Set the new cluster as the current best
      u ← MaxTrials(size($\mathcal{L}$), size($\mathcal{S}$),s) // Update the required number of trials
    $k ← k + 1$
  until $k > u$ or size($\mathcal{L}$) $\leq s$
  $\mathcal{L} ← \mathcal{L} \backslash \mathcal{S}$ // Remove the dominant cluster from the set of axes
Function SampleKeyholePosition
  Input: $\mathcal{L} = \{L_1, ..., L_m\}, f, s$ // Set of 3D axes, maximum distance for valid supporting axis, minimum cluster size
  Output: $\mathcal{L}, x$ // Set of remaining 3D axes, 3D keyhole location
  found ← false
  // Main sampling loop
  while not found and size($\mathcal{L}$) $\geq s$ **do**
    A ← SampleOneAxisRandomly($\mathcal{L}$) // Random sample the initial axis
    [found, x] ← FindIntersectingAxisRandomly($\mathcal{L}, A, f$) // Attempt to find an axis that intersects the initial one
    if not found **then**
      $\mathcal{L} ← \mathcal{L} \setminus A$ // If no intersecting axis is found, remove the initial axis from the set of axes
  end

Algorithm 1: Estimation of 3D keyhole locations through clustering of tool 3D axes.
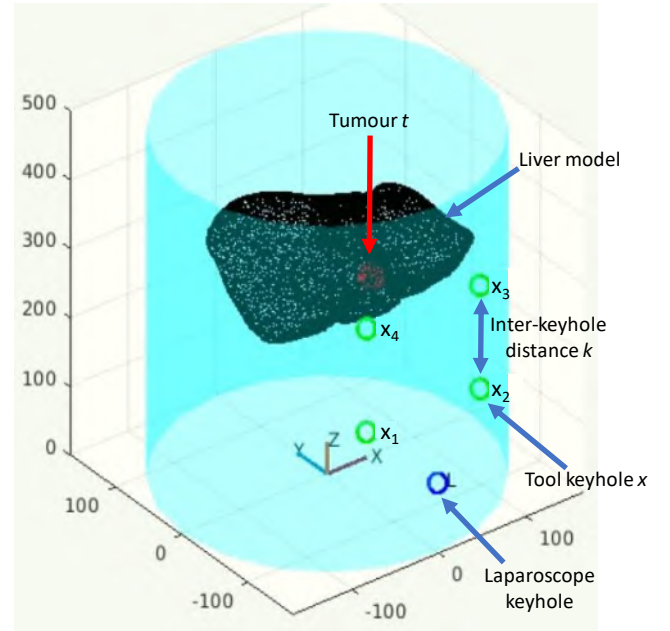

Fig. 4: Geometric in silico 3D model of the laparoscopic settings used to generate semi-synthetic data.
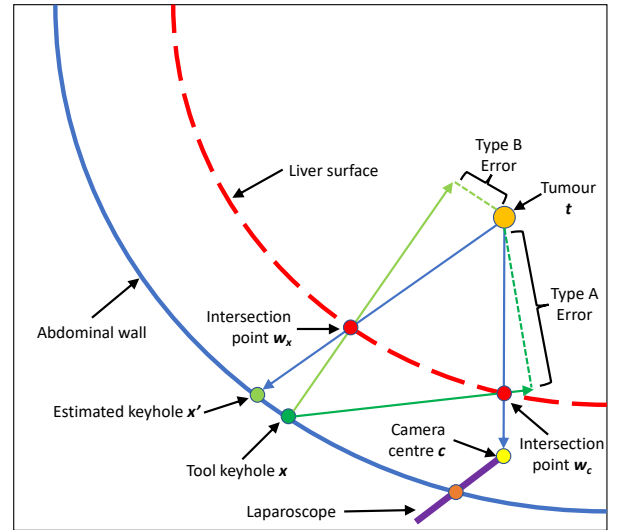

Fig. 5: Procedure for measuring pointing errors. The Type-A error is the distance between the tumour t and the tool axis pointing to the projection w_c of t on the liver surface towards the camera centre c. Type-B error is the distance between the tumour t and the tool axis pointing to the projection w_c of t on the liver surface towards the estimated keyhole x'.

depth as the distance between the tumour's centre and the liver's anterior face. We vary this depth by moving the tumour along the anterior-posterior liver axis. At every step, we estimate a keyhole position x' and measure the pointing errors 100 times. The results for these pointing errors are presented in section 4.1.

3.2. Phantom setup

We design a realistic experimental pipeline that uses one of the 3D-printed liver phantoms generated in Espinel et al. (2021), along with a STORZ monocular endoscopy system. A complete view of the setup is shown in figure 6. As for the simulated environment case, the goal of this setup is to measure the keyhole estimation errors and the pointing errors by using the estimated keyholes. A desktop PC with the AR software Hepataug Koo et al. (2017); Espinel et al. (2020) is connected to the video output port of the endoscopy system. The liver phantom is positioned inside a pelvitrainer and the laparoscope is fixed using a Martin's arm holder, in a way that gives a global view of the phantom. We use three keyholes, as shown in figure 7(a). To compute the keyhole errors, the groundtruth keyhole locations

are found from the pose of a special ChArUco board in the laparoscopic image, as shown in figure 7(b), where the marker is glued to a house-shaped piece with known size, whose tip is positioned at the keyhole. To measure the pointing errors, an external camera is mounted on a tripod, looking towards the pelvitrainer and the liver phantom. A ChArUco board is also attached to a surgical tool, in order to find the tool poses during the pointing experiments, as shown in figure 7(c). Both the laparoscope and the external camera were properly calibrated using a checkerboard pattern. Concretely, the checkerboard pattern was filmed using each camera. Then, selected images from

each video sequence were imported in the Metashape software developed by Agisoft LLC (2023), which then estimated the intrinsic camera parameters.

Assessment of the clustering success and keyhole error. Three surgical tools were inserted simultaneously through the aforementioned keyholes, and a video stream was captured with the tools making circular movements. From the captured video, 30 frames were selected and the tool primitives were manually annotated for each frame. An example of the extracted frames is shown in figure 8(a), with the corresponding primitives shown in figure 8(b). Then, the 3D tool poses were estimated from the annotated primitives using the method from Hasan et al. (2021), resulting in a set of 30 unordered tool poses as shown in figure 8(c). The results for the clustering success and the keyhole estimation errors on this set of tool poses are shown in section 4.2.

Assessment of pointing errors. After estimating the keyholes from the previous set of tool poses, we perform keyhole-aware AR on the liver phantom and measure the pointing errors on virtual tumours. A preoperative 3D model of the liver phantom containing 8 virtual tumours is registered to the laparoscopic image using the method from Koo et al. (2017). In order for the user to visually assess the registration quality, the registered liver surface mesh model is overlaid onto the surgical image. Then, we ask 13 volunteers to point with the surgical tool from every keyhole towards each of the tumours using direct camera projection and keyhole aware projections. This means that, if the instrument is inserted through keyhole x_1, they pointed to the camera projection of a tumour, and to the projection of that tumour towards the estimated location of x_1. An example of such pointings is shown in figure 9. For every pointing experiment, we take a photo of the tool and the phantom with the external camera. This gives a total of 624 pointings, where each volunteer made 24 pointings using the camera-based AR and 24 pointings using the keyhole-based AR. To measure the pointing errors, we rigidly register the 3D model used to print the liver phantom, which also contains the groundtruth locations of the deformed tumours, to the corresponding external camera image using PnP. We also find the pose of the surgical tool in every image from the attached ChArUco board. The number of successfully reconstructed tool poses and the pointing errors are presented in section 4.2.

3.3. Clinical cases

We tested our method qualitatively on three patients, for which the surgeon fixed the laparoscope using a Martin's arm holder. We followed an IRB approved protocol HERAP2 from CPP Sud-Est VI, 2022/CE23. In the first patient, the surgeon opened 5 keyholes in the abdominal cavity: 4 for the surgical tools and 1 for the laparoscope. In the second patient, the surgeon opened 3 keyholes: 2 for the surgical tools and 1 for the laparoscope. In the third patient, the surgeon opened 4 keyholes: 3 for the surgical tools and 1 for the laparoscope. The keyhole placements are shown in figure 10. For each patient, the surgeon performed camera calibration and started moving the tools while the video stream was being recorded. We estimated the tool primitives, the tool poses, and the keyholes for

each patient in a retrospective manner. We report these results in section 4.3.

4. Results and Discussion

4.1. Geometric in silico model

Clustering success and keyhole estimation error. For each variation of the evaluation parameters, we generated boxplots for the clustering success and the keyhole error from the 400 experiments ran on the four keyholes $\{x_1, x_2, x_3, x_4\}$. These boxplots are shown in figures 11 for the clustering success and in figure 12 for the keyhole error. In all the boxplots, the whisker length is set as $\beta = \gamma_3 + 1.5(\gamma_3 - \gamma_1)$, where $\{\gamma_1, \gamma_3\}$ are the 25th and 75th percentiles of the data respectively. We observe that increasing the tool axis noise decreases clustering success. Starting at a nearly perfect median of 98.5%, with a maximum of 100% and a minimum of 90% without noise, it decreases to a median of 62.6%, with a maximum of 71.1% and a minimum of 52.2% at 20 mm noise. In spite of this decrease in clustering success, we observe that the keyhole error, starting at a median of 1.9 mm, with a maximum of 5.5 mm and a minimum of 0.3 mm, increases to a median of 3.9 mm only, with a maximum of 15.2 mm and a minimum of 0.6 mm. As expected, increasing the noise magnitude also increases the spread of both the clustering success and the keyhole errors. Increasing the number of observed tools does not affect the average clustering success but dramatically reduces its spread, from a maximum of 100% and a minimum of 83.3% for 100 tools to a maximum of 99.2% and a minimum of 89.4% for 2000 tools, while substantially decreasing the keyhole error from a median of 3.7 mm, with a maximum of 16.9 mm and a minimum of 0.4 mm, to a median of 1.5 mm, with a maximum of 3.9 mm and a minimum of 0.1 mm, stabilising beyond 1500 observed tool axes, representing a minute of video at 25 fps. Varying the percentage of erroneous tools from 0% to 50% does not affect the clustering accuracy significantly either, remaining at a satisfying rate of 96.5%, with a maximum of 100% and a minimum of 83.3% for a rate of erroneous tools of 40%. On the other hand, the keyhole error increases from a median of 1.5 mm, with a maximum of 5.8 mm and a minimum of 0.2 mm, to a median of 3.5 mm, with a maximum of 14.5 mm and a minimum of 0.1 mm. It is worth noting that most of the maximum and minimum values, represented as red crosses in the boxplots, correspond to isolated cases that are far from the rest of the data.

Assessment of the pointing errors. From the pointing errors reported in figure 13, we observe that the type-A error is always larger than the type-B error. The type-A error is strongly affected by the tumour depth, with a critical error beyond 35 mm. The variations are due to the location of keyhole x_1, the liver's irregular shape, and its position inside the cylinder modelling the patient's abdomen. We illustrate this in figure 14. At a depth of 5 mm, the tumour is close to the liver's anterior surface and the intersection points $\{w_c, w_x\}$ are close to each other, making the camera-based and keyhole-based pointings to be both close to the tumour. At a depth of 50 mm, the tumour is located in the middle of the liver and the intersection points $\{w_c, w_x\}$ are far from each other, making the camera-based pointing to
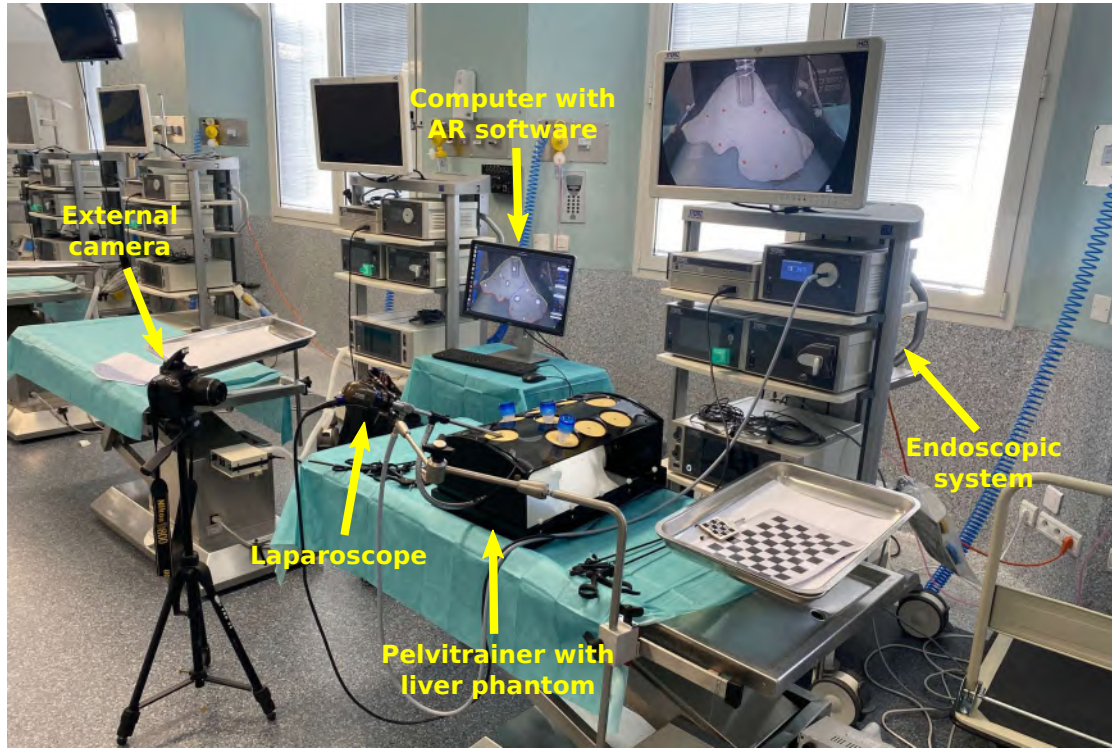
Fig. 6: Setup for keyhole estimation experiments using phantom data. Left: the external camera used to measure the pointing errors. Centre: the pelvitrainer with the liver phantom and the laparoscope fixed with a mechanical arm. Right: the PC with the AR software connected to the endoscopy system.
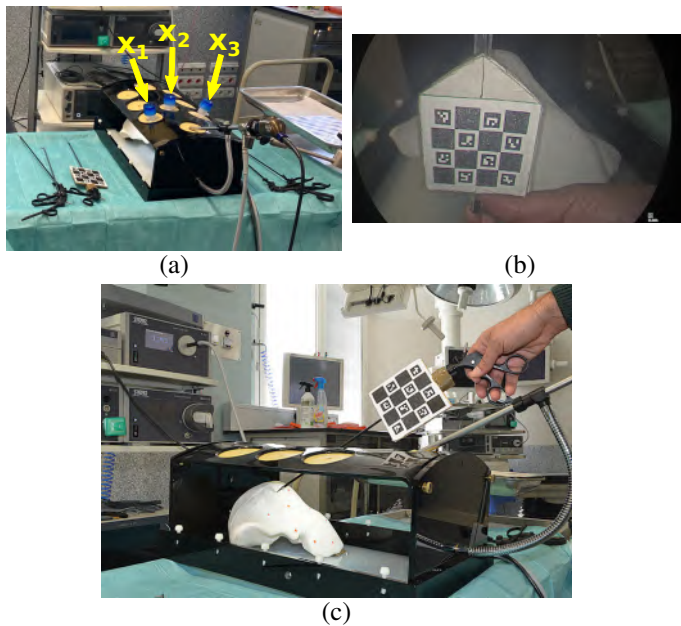

Fig. 7: Details of the phantom experimental setup. (a) The 3 keyholes used in the experiments; (b) Laparoscopic image showing a ChArUco board to estimate a groundtruth keyhole location; (c) View from the external camera showing a tool with the attached ChArUco board pointing towards the liver phantom.
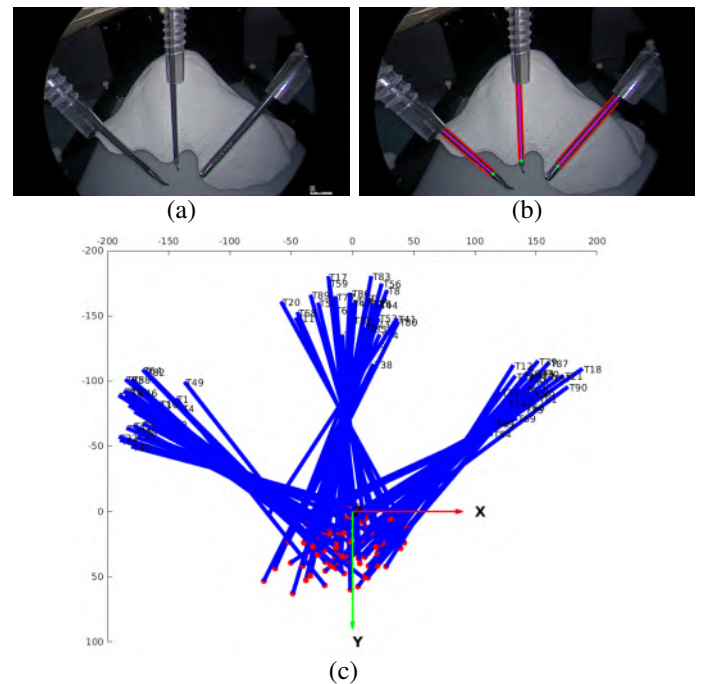

Fig. 8: Tools used to estimate the keyhole positions. (a) Laparoscopic image of the tools being inserted in the 3 keyholes; (b) Annotated tool primitives corresponding to the edge lines, middle line, and tool tip; (c) Reconstructed tool poses in laparoscope's reference frame

be far from the tumour, yet keeping the keyhole-based pointing accurate. At a depth of 100 mm, the tumour is close to the liver's lower-posterior surface and, because this region is visible to both the laparoscope and the tool keyholes, the intersection points $\{w_c, w_x\}$ are close to each other again, making both
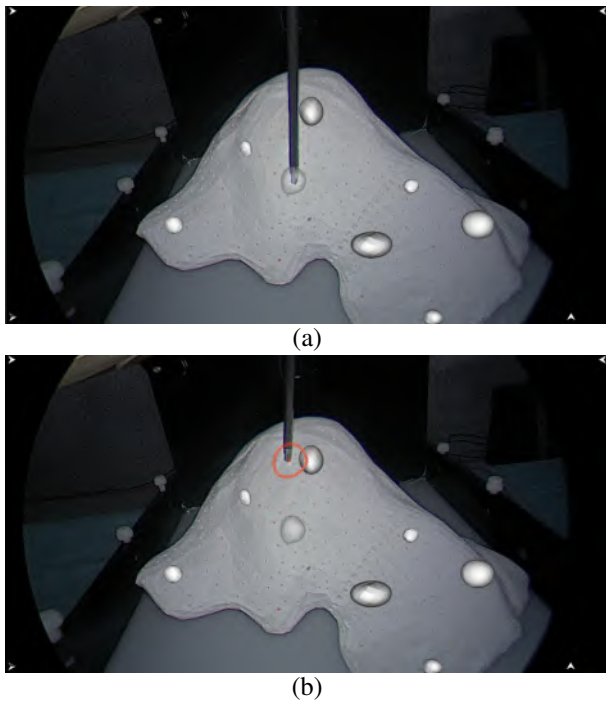
(a)

(b)

Fig. 9: Example of tumour pointing using AR on a liver phantom: (a) Pointing to the camera-based projection of the tumour; (b) Pointing to the keyhole-based projection of the tumour.

(a)

(b)

(c)

Fig. 10: Placement of keyholes and laparoscope for first patient (a), second patient (b), and third patient (c).
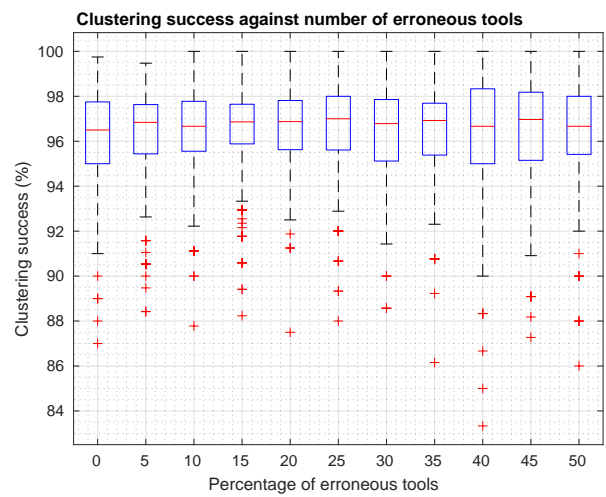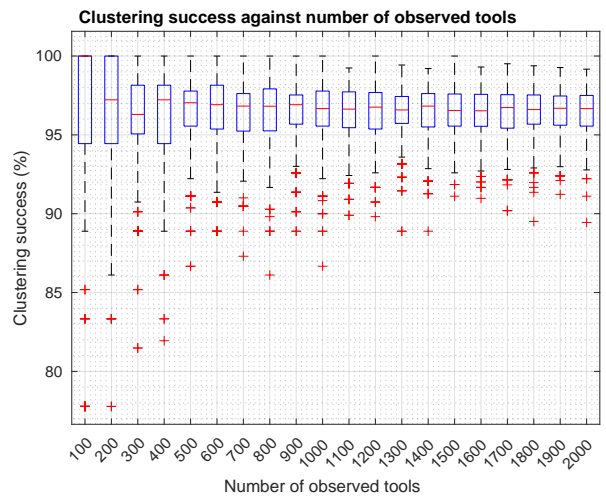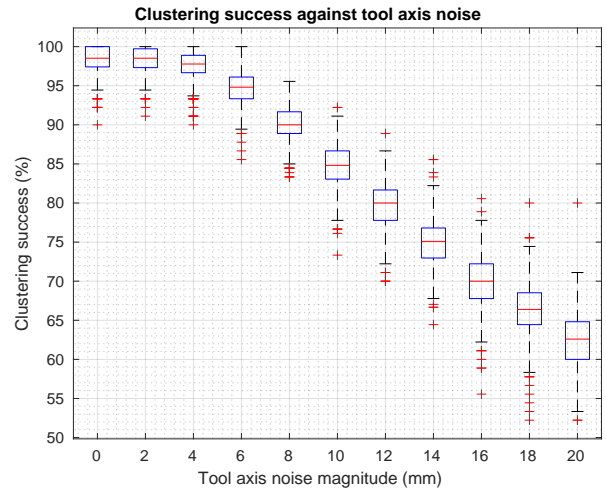

(a)

(b)

(c)

Fig. 11: Clustering success against (a) tool axis noise, (b) number of observed tool axes, and (c) percentage of erroneous tool axes.
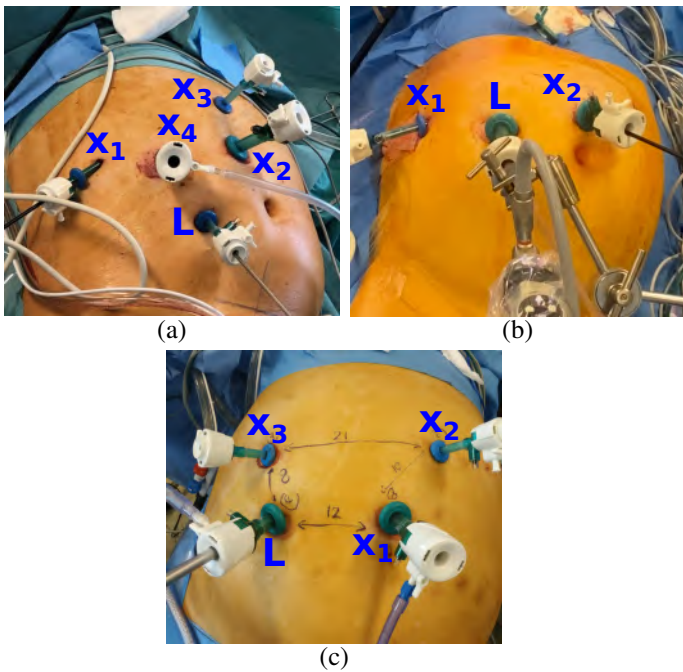
pointings accurate. This is however an example of configuration. In practice, other configurations also occur, in particular with a posterior surface invisible from the camera, causing the camera-based pointing error to strictly increase with depth. In
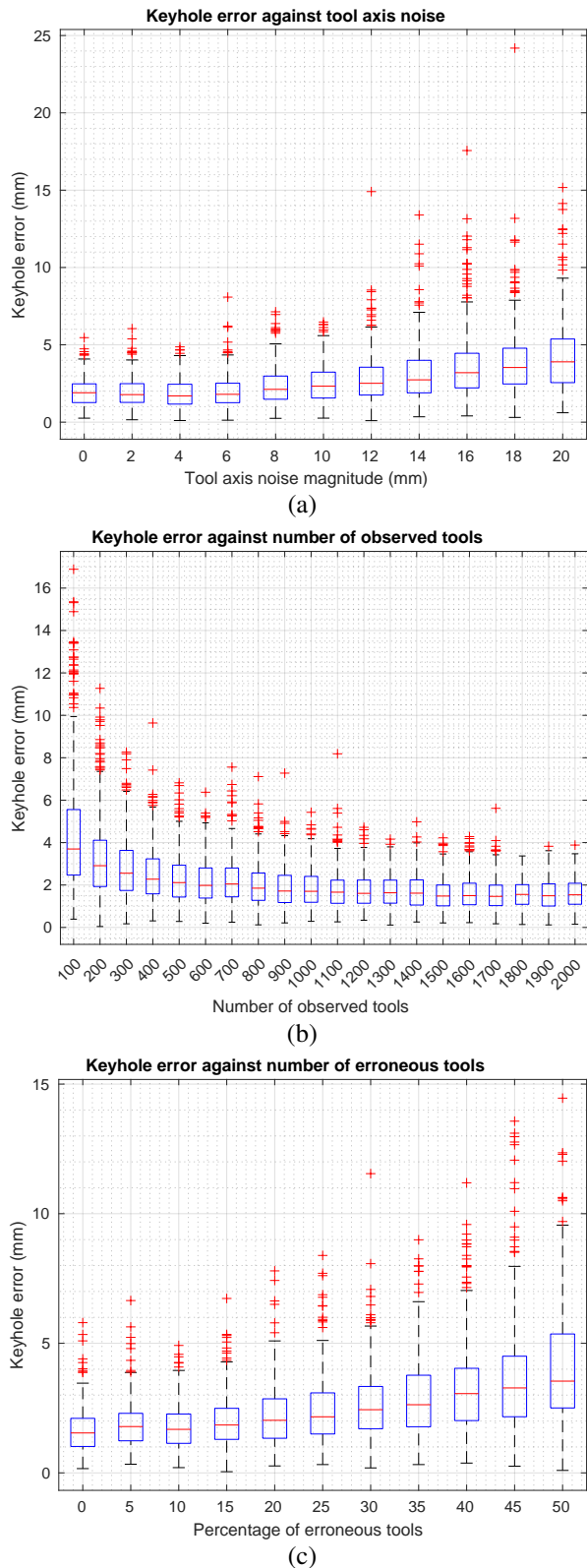
Fig. 12: Keyhole error against (a) tool axis noise, (b) number of observed tool axes, and (c) percentage of erroneous tool axes.

contrast the type-B error is not significantly affected by tumour depth, being always lower than 1 mm. The type-A error linearly increases with inter-keyhole distance, from 21.16 mm to 37.74

mm. This was expected, as closer tool and camera keyholes reduce the direct visualisation bias. The type-B error is unaffected, remaining at 0.48 mm. The type-A error is obviously unaffected by keyhole uncertainty, as it does not depend on the keyhole position, while the type-B error linearly increases from 0.36 mm to 0.79 mm, remaining largely lower than the type-A error.

4.2. Phantom setup

Clustering success and keyhole estimation error. We use our method to cluster the set of tool poses from section 3.2 and estimate the keyhole locations. By setting the global parameters $s = 10$ and $f = 10$ mm, we were able to cluster the 3 keyholes and estimate their locations in the laparoscope's coordinate frame, as shown in figure 15(b). The number of clustered tools per keyhole and the errors on the estimated keyhole locations are shown in table 1. The average keyhole position error was 19.7 mm. Importantly, the keyhole is affected by two types of errors: modelling errors and estimation errors. Modelling errors stem from the mathematical model of a keyhole as a static point; in reality however, a keyhole is neither static, as the abdominal wall has an extent of flexibility, nor a point, as a keyhole has a spatial extent. An intuition that may arise is that the keyhole can move due to the surgeon pushing the tool inside the cavity, making the abdomen to deform along the tool's axis, or due to moving the tool in different directions, making the abdomen to deform perpendicularly to the tool's axis. To confirm in which direction the keyhole moves the most, we have performed an experimental analysis in order to provide numerical evidence regarding this intuition. We computed a dominant tool axis L_d for each of the estimated keyholes $\{x'_1, x'_2, x'_3\}$ as the directional average of the axes in cluster $\mathcal{S}'_i$. We then projected the keyhole error vector $E = x_i - x'_i$ onto L_d, obtaining vector E_L, and onto the plane tangent to L_d and passing by the keyhole x'_i, obtaining vector E_P. Finally, we measured the magnitudes of both E_L and E_P, obtaining respectively the in-axis and the tangent errors, which provide estimates to the in-depth and lateral errors of the keyhole estimates. For keyhole x'_1 we obtained $\|E_L\| = 1.94$ mm and $\|E_P\| = 21.77$ mm; for keyhole x'_2 we obtained $\|E_L\| = 11.81$ mm and $\|E_P\| = 8.56$ mm; for keyhole x'_3 we obtained $\|E_L\| = 2.55$ mm and $\|E_P\| = 22.55$ mm. We do not observe a clear trend. Quantitatively however, the lateral error is larger than the in-depth error, and fails to confirm the above discussed intuition. On the other hand, estimation errors stem from the estimation pipeline. They can be strong, as the keyhole is never observed directly by the camera and can thus only be estimated indirectly from observations of the tool axes. The error in estimating the tool axes depends on two main factors, the errors from camera calibration and the errors from image-based tool primitive detection by ART-Net. The final estimate for the keyhole position critically depends on the number of tool axes being involved. In practice, only the estimation error sources can be acted upon, without incurring profound model and methodological changes. In our experiments, we have carefully calibrated the camera from 117 images selected where the checkerboard was sharp and covered the whole image. This allowed us to obtain a sufficiently accurate estimate of the camera
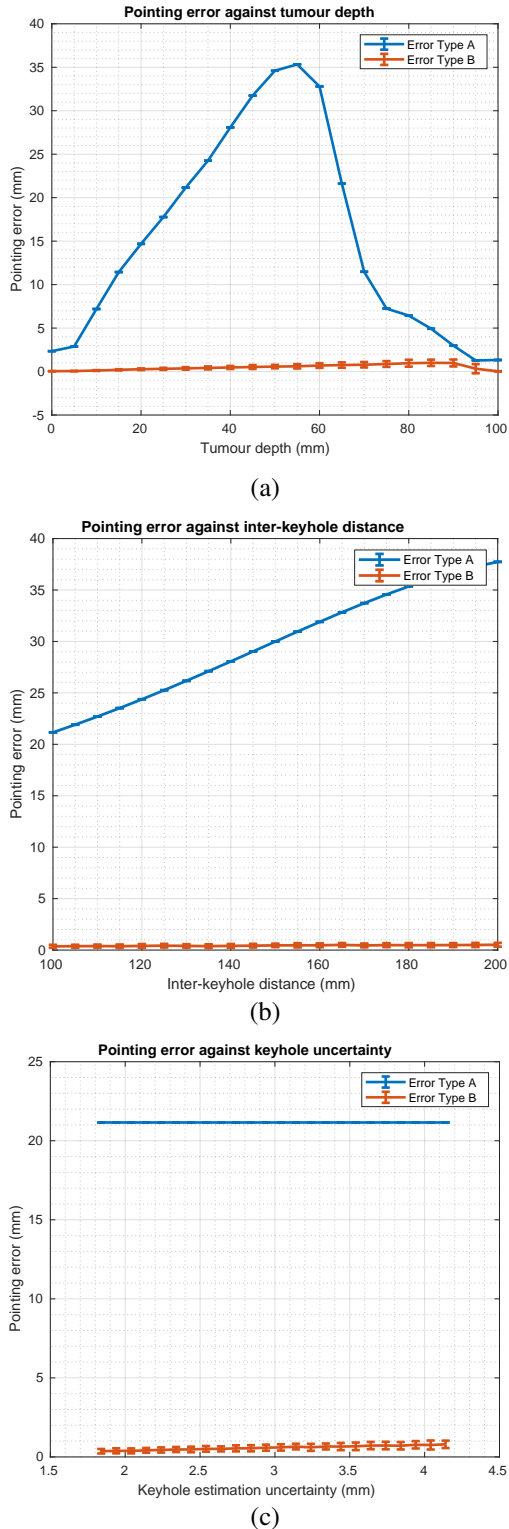
(a)

(b)

(c)

Fig. 13: Type A-and type-B pointing errors against tumour depth (a), inter-keyhole distance (b), and keyhole precision (c).

parameters. According to the results presented by Hasan et al. (2021), the estimated tool poses may have angular errors up to 9° and positional errors up to 6 mm. In these phantom experiments, we could not use ART-Net's automatic tool primitive detection, as it was never trained to cope with phantom images.

Table 1: Clustering results and keyhole estimation errors.

Keyhole	# of detected tools	Keyhole estimation error (mm)
x_1	27	21.9
x_2	25	14.6
x_3	18	22.7
Average	**22**	**19.7**

We thus annotated the tool primitives manually, which limited the number of images that we could use to estimate the tool poses to 30 images per keyhole. However, as shown in section 4.1, a lower bound on the number of tool poses required to accurately estimate a keyhole is in the order of 1000. While this can be reasonably reached for clinical data, for which ART-Net automatically processes the images, this is not feasible for manually annotated phantom images. To sum up, the phantom model has a lower model error than the patient model as it undergoes more restricted keyhole motion but has a higher estimation error as it can only be estimated with a restricted number of tool poses. All in all, it is thus a challenging setup, providing a non-trivial validation of the proposed method.

Assessment of pointing errors. Some of the pointing images captured in section 3.2 were not suitable for obtaining valid 3D tool poses, owing to occlusions of the ChArUco board from the external camera; the number of pointing images with valid tool poses per tumour and per keyhole is given in table 2. Finally, we compute the average type-A error for the camera projections and the average type-B error for the keyhole projections, by measuring the distance between every tumour centre and the tool axes. These errors are reported in table 3. We observe a reduction of 16.61 mm in the pointing error when using the keyhole-aware AR compared to the existing direct camera projection AR. It is worth noticing that the pointing errors include inaccuracies coming from the preoperative-to-intraoperative registration, the keyhole estimations, and the pointing gestures done by the user. For example, we can observe that the type-B errors per keyhole are proportional to the keyhole estimation errors, which is an expected outcome of the experiments. The single-view registration method Koo et al. (2017) works better with frontal views of the liver having a good visibility of the anterior anatomical landmarks. These landmarks exert weaker constraints on the posterior part of the liver, which may reduce the registration accuracy in this area. This shortcoming is addressed by the multiple-view registration method Espinel et al. (2021), which combines frontal and lateral views of the liver. Using this method could improve the registration accuracy and, therefore, reduce the pointing errors.

4.3. Clinical cases

We start by testing the performance of the pretrained ART-Net model from Hasan et al. (2021) on the patients introduced in section 3.3. We extracted 116 images from the video of the first patient, with 69 images corresponding to keyhole x_1, 19 to keyhole x_2, 7 to keyhole x_3, and 4 to keyhole x_4. For the second patient we extracted a total of 50 images, with 26 images corresponding to keyhole x_1 and 24 to keyhole x_2. For the third pa-
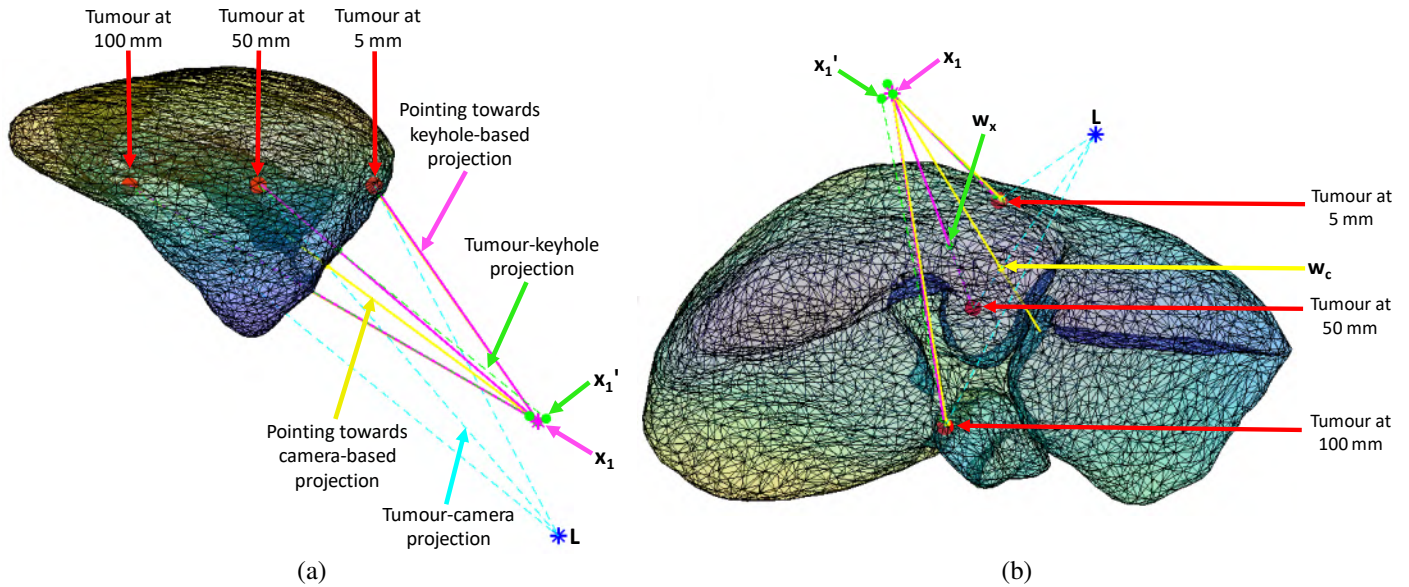
Fig. 14: Pointing experiments in the geometric in silico model for three tumour depths: (a) View from the liver's right lobe; (b) View from the liver's bottom side.
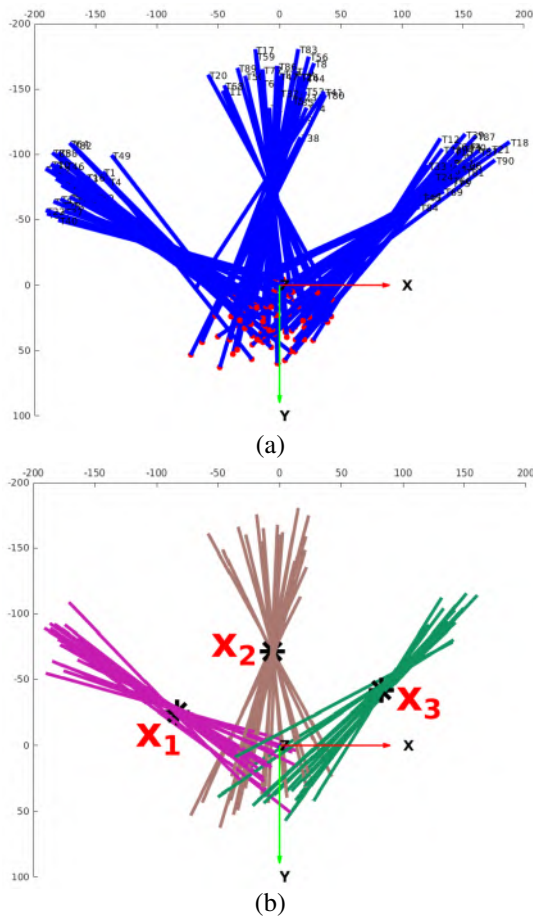

Fig. 15: Tool poses and estimated keyholes: (a) Reconstructed tool poses in laparoscope's reference frame; (b) Clustered tools and estimated keyholes.

Table 2: Number of pointing images with valid tool poses for 13 volunteers using camera-based projections (a) and keyhole-based projections (b).

Keyhole → Tumour ↓	x_1	x_2	x_3	Total
t_1	11	13	6	**30**
t_2	13	13	7	**33**
t_3	12	13	13	**38**
t_4	13	13	0	**26**
t_5	13	13	13	**39**
t_6	3	13	13	**29**
t_7	1	13	13	**27**
t_8	5	12	13	**30**
Total	**71**	**103**	**78**	**252**

(a)

Keyhole → Tumour ↓	x_1	x_2	x_3	Total
t_1	11	13	6	**30**
t_2	9	12	11	**32**
t_3	12	13	10	**35**
t_4	11	12	1	**24**
t_5	13	13	13	**39**
t_6	3	13	13	**29**
t_7	0	13	13	**26**
t_8	5	13	13	**31**
Total	**64**	**102**	**80**	**246**

(b)

tient we extracted 43 images, with 20 images corresponding to keyhole x_1, 10 to keyhole x_2, and 13 to keyhole x_3. We observed a poor performance in detecting the tool primitives, which we explain by the fact that ART-Net was originally trained on a limited dataset with a different distribution compared to our images. We thus fine-tuned ART-Net with 200 additional images from 11 liver surgeries which we manually annotated. These 11

Table 3: Pointing errors for camera projections (type A) and keyhole projections (type B).

Keyhole	Type-A error (mm)	Type-B error (mm)
x_1	22.0 ± 12.5	09.5 ± 06.5
x_2	26.6 ± 14.0	11.6 ± 10.9
x_3	50.5 ± 21.4	27.1 ± 12.6
Average	**32.7 ± 20.3**	**16.1 ± 13.0**

patients are different from the 3 test patients we use to estimate the tool poses and the keyholes. We augmented the training data by applying random flipping, scaling, blurring and rotation. The fine-tuning improved ART-Net's results as shown in figure 16, and we could obtain the geometric primitives and the tool 3D poses for most of the images. For the first patient, we obtained reasonable tool poses in 56 images for keyhole x_1, 16 images for keyhole x_2, 5 images for keyhole x_3, and 16 images for keyhole x_4. For the second patient, we obtained valid tool poses in 22 images for keyhole x_1 and 24 images for keyhole x_2. For the third patient, we obtained valid tool poses in 19 images for keyhole x_1, 9 images for keyhole x_2, and 11 images for keyhole x_3. We then used our method to estimate the corresponding keyhole locations, and proceed to project the tumours towards the keyholes. Renderings of the tumour projections towards the estimated keyholes are shown in figure 17 for the first patient, figure 18 for the second patient, and figure 19 for the third patient. We observe from the keyhole-aware renderings that the tumour projections are oriented towards the entry points of the tools. The slight deviation is explained by the fact that the surgeon was not specifically pointing at the tumour, hence only the keyhole direction is relevant to observe. In our protocol, keyhole locations were estimated after surgery.
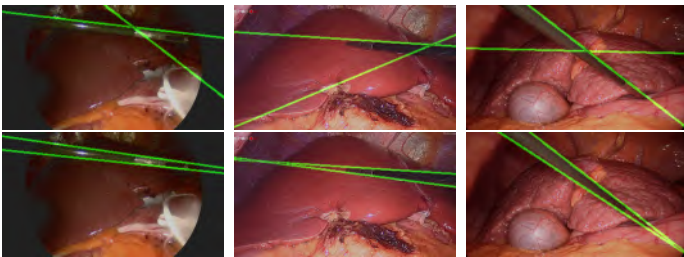

Fig. 16: Detection of tool edge primitives before (top) and after (bottom) fine-tuning of ART-Net.

5. Conclusion

We have presented a visualisation method for surgical AR in laparoscopic liver surgery, which takes into account the location of the keyholes of the tools used to resect internal tumours. Specifically, we have proposed keyhole-aware visualisation, where the tumour is not directly rendered to the image plane, but first projected to the liver surface towards the aiming keyhole. For that, we have proposed to estimate the keyhole position by reconstructing the tool 3D pose for several images and computing their shaft axis intersection point. Our method copes with noise and erroneous tool axes. We have shown that
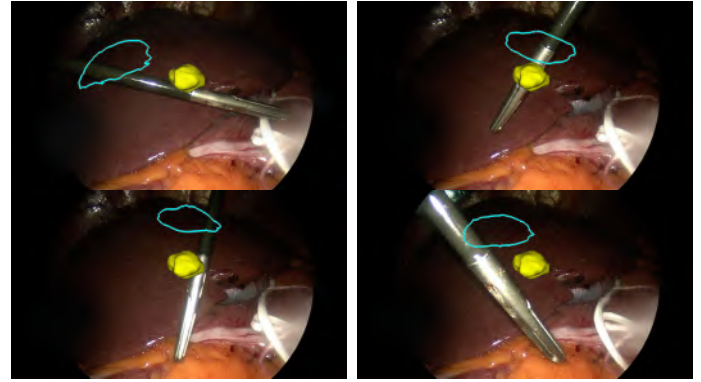
Fig. 17: Keyhole-aware AR with tumour projection towards the estimated keyhole x_1 (top-left), keyhole x_2 (top-right), keyhole x_3 (bottom-left), and keyhole x_4 (bottom-right) of the first patient.
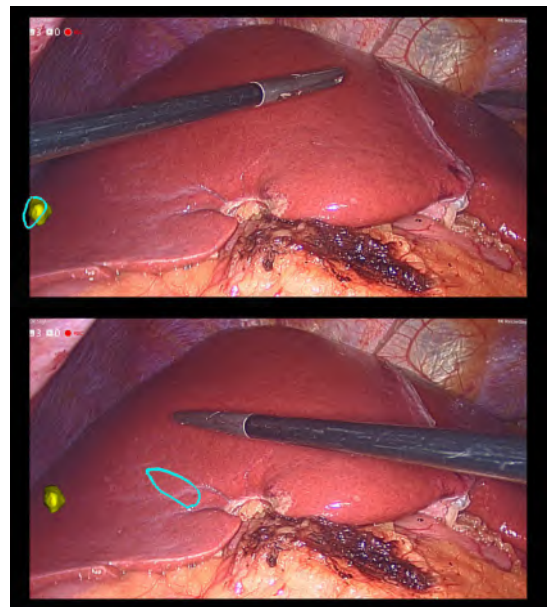
Fig. 18: Keyhole-aware AR with tumour projection towards the estimated keyhole x_1 (top) and keyhole x_2 (bottom) of the second patient.
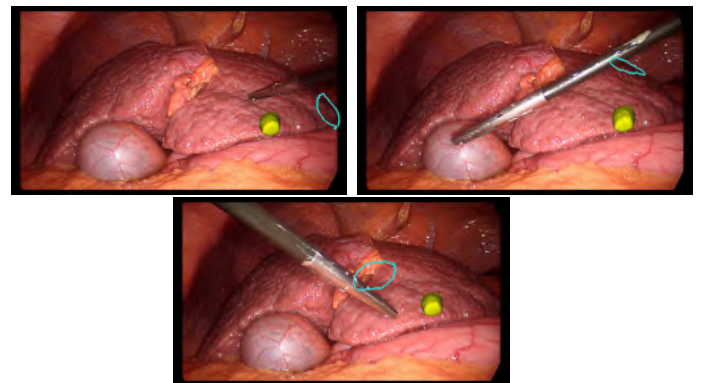
Fig. 19: Keyhole-aware AR with tumour projection towards the estimated keyhole x_1 (top-left), keyhole x_2 (top-right), and keyhole x_3 (bottom) of the third patient.

keyhole-aware visualisation fixes a gap in existing AR systems and may have a dramatic impact on AR usability and surgical

safety. Compared to existing direct camera projection methods, which provides misleading guidance information on the tumour's real position, keyhole-aware rendering makes it possible for the surgeon to directly aim at the tumour with the surgical tool. This could increase the precision of laparoscopic resection and reduce patient trauma.

We plan to extend our system in four ways. First, we plan to handle a mobile organ such as the kidney, in which case, the keyholes must be dynamically updated from new tool observations available during resection. Second, we plan to handle the presence of multiple tools visible at once, for which the tool 3D pose method which we re-used from the literature will have to be extended with an initial step of instance segmentation. Third, we plan to take into account the vascularity around the tumour and its location inside the liver to generate a suitable resection path. Fourth, we plan to extend the clinical evaluation of our method, by quantifying the effect in terms of depth perception brought by the various visualisation modes. This includes assessing the benefits of using the projection cylinder, the resection path and collecting feedback from surgeons to improve their design.

Ethical Approval

Acknowledgements

Author contributions

Yamid Espinel: Investigation, Methodology, Software, Data curation, Validation, Writing - Original draft. **Navid Rabbani:** Methodology, Software, Formal analysis, Validation. **Thien Bao Bui:** Software, Data curation, Validation. **Mathieu Ribeiro:** Methodology, Resources. **Emmanuel Buc:** Investigation, Resources. **Adrien Bartoli:** Conceptualisation, Methodology, Formal analysis, Supervision, Writing - Review and editing.

References

Adagolodjo, Y., Trivisonne, R., Haouchine, N., Cotin, S., Courtecuisse, H., 2017. Silhouette-based pose estimation for deformable organs application to surgical augmented reality, in: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 539–544.

Agisoft LLC, 2023. Agisoft Metashape. URL: https://www.agisoft.com/.

Bernhardt, S., Nicolau, S., Bartoli, A., Agnus, V., Soler, L., Doignon, C., 2015. Using shading to register an intraoperative ct scan to a laparoscopic image, in: Medical Image Computing and Computer Assisted Intervention – MICCAI 2015.

Buell, J.F., Cherqui, D., Geller, D.A., O'Rourke, N., Iannitti, D.A., Dagher, I., Koffron, A.J., Thomas, M.J., Gayet, B., Han, H.S., Wakabayashi, G., Belli, G., Kaneko, H., Ker, C.G., Scatton, O., Laurent, A., Abdalla, E.K., Chaudhury, P.K., Dutson, E.P., Gamblin, C.T., D'Angelica, M.I., Nagorney, D.M., Testa, G., Labow, D.M., Manas, D., Poon, R.T.P., Nelson, H., Martin, R.C.G., Clary, B.M., Pinson, W.C., Martinie, J.B., Vauthey, J.N., Goldstein, R.M., Roayaie, S., Barlet, D., Espat, J., Abecassis, M.M.I., Rees, M., Fong, Y., McMasters, K.M., Broelsch, C.E., Busuttil, R.W., Belghiti, J., Strasberg, S.M., Chari, R.S., 2009. The international position on laparoscopic liver surgery: The louisville statement, 2008. Annals of Surgery 250, 825–830.

Chen, L., Tang, W., John, N., Wan, T., Zhang, J., 2018. Slam-based dense surface reconstruction in monocular minimally invasive surgery and its application to augmented reality. Computer Methods and Programs in Biomedicine 158.

Cheung, T.T., Poon, R.T.P., Yuen, W.K., Chok, K.S.H., Tsang, S.H.Y., Yau, T., Chan, S.C., Lo, C.M., 2013. Outcome of laparoscopic versus open hepatectomy for colorectal liver metastases. ANZ Journal of Surgery 83, 847–852.

Clements, L., Collins, J., Weis, J., Simpson, A., Kingham, T., Jarnagin, W., Miga, M., 2017. Deformation correction for image guided liver surgery: An intraoperative fidelity assessment. Surgery 162.

Collins, T., Chauvet, P., Debize, C., Pizarro, D., Bartoli, A., Canis, M., Bourdel, N., 2017. A system for augmented reality guided laparoscopic tumour resection with quantitative ex-vivo user evaluation, pp. 114–126.

Espinel, Y., Calvet, L., Botros, K., Buc, E., Tilmant, C., Bartoli, A., 2021. Using multiple images and contours for deformable 3d-2d registration of a preoperative ct in laparoscopic liver surgery, in: Medical Image Computing and Computer Assisted Intervention – MICCAI 2021, Springer International Publishing. pp. 657–666.

Espinel, Y., Özgür, E., Calvet, L., Le Roy, B., Buc, E., Bartoli, A., 2020. Combining visual cues with interactions for 3d–2d registration in liver laparoscopy. Annals of Biomedical Engineering 48, 1712–1727.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395.

Haouchine, N., Dequidt, J., Peterlik, I., Kerrien, E., Berger, M.O., Cotin, S., 2014. Towards an accurate tracking of liver tumors for augmented reality in robotic assisted surgery, in: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 4121–4126. doi:10.1109/ICRA.2014.6907458.

Haouchine, N., Roy, F., Untereiner, L., Cotin, S., 2016. Using contours as boundary conditions for elastic registration during minimally invasive hepatic surgery, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 495–500.

Hartley, R., Zisserman, A., 2004. Multiple View Geometry in Computer Vision. Cambridge University Press. URL: https://books.google.fr/books?id=eLbfoQEACAAJ.

Hasan, M.K., Calvet, L., Rabbani, N., Bartoli, A., 2021. Detection, segmentation, and 3d pose estimation of surgical tools using convolutional neural networks and algebraic geometry. Medical Image Analysis 70, 101994.

Koo, B., Özgür, E., Le Roy, B., Buc, E., Bartoli, A., 2017. Deformable registration of a preoperative 3d liver volume to a laparoscopy image using contour and shading cues, in: Medical Image Computing and Computer Assisted Intervention – MICCAI 2017, Springer International Publishing. pp. 326–334.

Le Roy, B., Ozgur, E., Koo, B., Buc, E., Bartoli, A., 2019. Augmented reality guidance in laparoscopic hepatectomy with deformable semi-automatic computed tomography alignment (with video). Journal of Visceral Surgery 156.

Malbrain, M., Roberts, D., Laet, I., De Waele, J., Sugrue, M., Schachtrupp, A., Duchesne, J., van Ramshorst, G., Keulenaer, B., Kirkpatrick, A., Ahmadi-Noorbakhsh, S., Mulier, J., Ivatury, R., Pracca, F., Wise, R., Pelosi, P., 2014. The role of abdominal compliance, the neglected parameter in critically ill patients - a consensus review of 16. part 1: Definitions and pathophysiology. Anaesthesiology intensive therapy 46, 392–405.

Özgür, E., Lafont, A., Bartoli, A., 2017. Visualizing in-organ tumors in augmented monocular laparoscopy, in: 2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct), pp. 46–51.

Plantefeve, R., Peterlik, I., Haouchine, N., Cotin, S., 2015. Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic

surgery. Annals of biomedical engineering 44.

Robu, M.R., Ramalhinho, J., Thompson, S., Gurusamy, K., Davidson, B., Hawkes, D., Stoyanov, D., Clarkson, M.J., 2018. Global rigid registration of ct to video in laparoscopic liver surgery. International Journal of Computer Assisted Radiology and Surgery 13, 947–956.

Thompson, S., Totz, J., Song, Y., Johnsen, S.F., Stoyanov, D., Ourselin, S., Gurusamy, K., Schneider, C., Davidson, B., Hawkes, D., Clarkson, M., 2015. Accuracy validation of an image guided laparoscopy system for liver resection. Proceedings of SPIE - The International Society for Optical Engineering 9415, 1–12.

Zhong, J.H., Peng, N.F., You, X.M., Ma, L., Xiang, X., Wang, Y.Y., Gong, W.F., Wu, F.X., Xiang, B.D., Li, L.Q., 2017. Tumor stage and primary treatment of hepatocellular carcinoma at a large tertiary hospital in china: A real-world study. Oncotarget 8, 18296–18302.